CLAIMS

What is claimed is:

1. A system for performing interleaved packet processing in a network router, wherein a packet to be routed includes a source address bit pattern and a destination address bit pattern that are each processed by a task processor in accordance with a data tree, said data tree including a plurality of nodes linked by branches wherein an instruction that is associated with each node within said data tree is utilized for determining which branch is to be taken in accordance with said source address bit pattern or said destination address bit pattern, said system comprising:

a first bank of registers for loading an instruction to be executed by said task processor at each node of said data tree in accordance with said source address bit pattern;

a second bank of registers for loading an instruction to be executed by said task processor at each node of said data tree in accordance with said destination address bit pattern; and

a task scheduler for enabling said first bank of registers to transfer an instruction loaded therein for processing by said task processor only during even time cycles and for enabling said second bank of registers to transfer an instruction loaded therein for processing by said task processor only during odd time cycles.

2. The system of claim 1, wherein said task scheduler includes a clock signal generator that generates said even and odd time cycles in an alternating series of rising edges and falling edges.

3. The system of claim 1, further comprising an address register for storing an address of a next instruction to be loaded into either said first bank of registers or said second bank of registers from a memory device before being executed by said task processor.

4. The system of claim 3, wherein said address register further comprises a counter for incrementing said address of the next instruction in response to a dual instruction.

5. The system of claim 3, wherein said memory includes instructions to be executed by said task processor.

6. The system of claim 5, wherein said memory further comprises a first memory area containing normal size instructions and a second memory area containing dual size instructions.

7. The system of claim 1, further comprising at least one temporary register for storing information from said task processor between two consecutive processing time cycles when such a processing lasts more than one time cycle.

8. The system of claim 8, further comprising a 1-bit state register for each of said first and second bank of registers, said 1-bit state register being set when said processing lasts more than one time cycle.

9. A method for performing interleaved packet processing in a network router, wherein a packet to be routed includes a source address bit pattern and a destination address bit pattern that are each processed by a task processor in accordance with a data tree, said data tree including a plurality of nodes linked by branches wherein an instruction that is associated with each node within said data tree is utilized for determining which branch is to be taken in accordance with said source address bit pattern or said destination address bit pattern, said method comprising:

loading into a first bank of registers an instruction to be executed by said task processor at each node of said data tree in accordance with said source address bit pattern;

loading into a second bank of registers an instruction to be executed by said task processor at each node of said data tree in accordance with said destination address bit pattern;

transferring an instruction from said first bank of registers to be processed by a task processor only during even time cycles; and

transferring an instruction from said second bank of registers to be processed by said task processor only during odd time cycles.

10. The method of claim 9, further comprising generating said even and odd time cycles in an alternating series of rising edges and falling edges.

11. The method of claim 9, further comprising storing an address of a next instruction to be loaded into either said first bank of registers or said second bank of registers from a memory device before being executed by said task processor.

12. The method of claim 11, further comprising:

loading said address of said next instruction from said task processor into said first or second bank of registers;

transferring said address from said bank of registers to said address register;

reading said address from said address register; and

fetching said next instruction from said memory in response to said reading step.

13. The method of claim 11, further comprising incrementing said address of the next instruction in response to a dual instruction.

14. The method of claim 13, further comprising:

loading a dual instruction into either said first bank of registers or said second bank of registers; and

interrupting said loading step during one time cycle if said loading requires two time cycles; and

during said time cycle during which said loading is interrupted, loading an instruction into the other of said first or second bank of registers.

15. The method of claim 13, further comprising:

processing a dual instruction utilizing said task processor;

interrupting said processing step during one time cycle if such a processing requires two time cycles; and

during said time cycle during which said processing is interrupted, executing an instruction provided by the other of said first or second bank of registers.

16. The method of claim 9, further comprising storing information from said task processor between two consecutive processing time cycles when such a processing lasts more than one time cycle within at least one temporary register.